REMARKS

Claim 1 calls for shutting down a television capture card when a crash is detected. It is conceded that two of the cited three references do not teach such a feature. Instead, it is suggested that Reinhardt teaches such a feature. However, the description of Reinhardt, on page 4, has nothing about doing anything when a crash is detected and has nothing to do with shutting down a television capture card.

To suggest that an error state due to high temperature is commensurate with a crash detection simply is not correct. A high temperature has nothing to do with a crash. In Reinhardt, high temperatures may be detected and, if the description on page 4 is correct, the computer shuts off power to the device to prevent damage when the temperature gets too high. But no crash was ever detected. No television capture card was ever shut down. Thus, the reference teaches absolutely nothing in any way pertinent to claim 1 and, therefore, the rejection should be reconsidered.

On the same basis, reconsideration of the rejections of claims 13, 21, 25, and 32 is respectfully requested.

Claim 28 calls for, in response to a request for video from a first application, initializing a video stream using a video server. If the first application crashes, access to the video stream is maintained for a second application to the video server and the server is directed to release the video stack.

However, nothing in Semenzato teaches maintaining access to the video stream even if the plug in crashes. Since it is the plug in which obtains the video and audio, it is not seen how access could be maintained.

Further, the material cited at column 5, lines 22-49, of Bopardikar relates to storing HDTV frames on a disk drive. There is no discussion of directing a server to release the video stack or a video stack of what happens when a first application crashes.

Also cited is column 26, line 23, through column 27, line 25. This material talks about the situation when a disk failure takes place. However, a disk failure has nothing to do with an application crashing, maintaining access to a video stream, or directing a server to release the video stack. Thus, a disk failure has nothing to do with an application failure.

2

Moreover, there is no suggestion of releasing a video stack since no video stack would be implemented. The system simply stores data on a disk. In a case of a failure of the disk, the data continues to be read out. It is explained in column 27 that since full frames of data are transferred to the router, the video environment is unaware of the disk failure. In such case, it would certainly be surprising that a video stack were released by the server.

Therefore, it is respectfully submitted that Bopardikar has nothing to do with the claimed invention since it relates to a non-analogous art of storing data on disk. It does not have the situation where an application crashes. It does not maintain access to the video stream for a second application through any server. It does not direct the server to release the video stack since the server is not involved in any disk operation.

Therefore, reconsideration of the rejection of claim 28 is respectfully requested. On the same basis, reconsideration of the rejection of claim 30 is requested.

Respectfully submitted,

Date: <u>September 28, 2005</u>

Timothy N. Trop, Reg. No. 28,994
TROP, PRUNER & HU, P.C.
8554 Katy Freeway, Ste. 100
Houston, TX 77024
713/468-8880 [Phone]
713/468-8883 [Fax]

Attorneys for Intel Corporation